6-2015

Method for matching probabilistic encrypted data

Hwee Hwa PANG
Singapore Management University, hhpang@smu.edu.sg

Xuhua DING
Singapore Management University, xhding@smu.edu.sg

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
18 June 2015 (18.06.2015)

WIPO | PCT

(10) International Publication Number
WO 2015/088448 A1

(51) International Patent Classification:
H04L 9/28 (2006.01) *G06F 21/10* (2013.01)

(21) International Application Number:
PCT/SG2014/000590

(22) International Filing Date:
10 December 2014 (10.12.2014)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
61/914,559 11 December 2013 (11.12.2013) US

(71) Applicant: **SINGAPORE MANAGEMENT UNIVERSITY** [SG/SG]; 81 Victoria Street, Singapore 188065 (SG).

(72) Inventors: **PANG, Hwee Hwa**; 201 Tanjong Rhu Road #15-11, Singapore 436917 (SG). **DING, Xuhua**; 15 Mount Emily Road, #04-38 Parc Emily, Singapore 228495 (SG).

(74) Agent: **ATMD BIRD & BIRD LLP**; 2 Shenton Way #18-01, SGX Centre 1, Singapore 068804 (SG).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK,

[Continued on next page]

(54) Title: METHOD FOR MATCHING PROBABILISTIC ENCRYPTED DATA

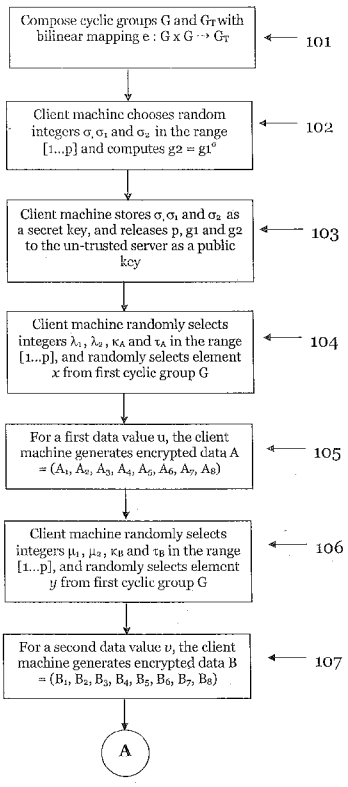

Figure 1

(57) Abstract: Determining if a first encrypted data of a first data value is equal to a second encrypted data of a second data value. Comprising: a first cyclic group; a second cyclic group including a first element. Applying an operation to the first cyclic group to map its elements to an element in the second cyclic group. Randomly selecting a second element from the first cyclic group; producing the first encrypted data by mapping the second element and the first data value into one or more elements of the first cyclic group. Randomly selecting a third element from the first cyclic group; producing the second encrypted data by mapping the third element and the second data value into one or more elements of the first cyclic group. Applying the operation to the first encrypted data and the second encrypted data to obtain a fourth element in the second cyclic group, wherein the fourth element is equal to the first element when the first data value is equal to the second data value.

WO 2015/088448 A1

SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

METHOD FOR MATCHING PROBABILISTIC ENCRYPTED DATA

FIELD OF THE INVENTION

[0001] The present invention relates to a new method for matching probabilistic encrypted data.

BACKGROUND

[0002] Typically in Information Technology (IT) outsourcing, a user may delegate the data storage and query processing functions to an un-trusted third-party server. This gives rise to the need to safeguard and ensure the privacy of the database as well as the user queries being sent to the database.

[0003] A method of preserving data privacy is by applying a deterministic encryption scheme to the data record values before storing them in the un-trusted servers. Therefore the un-trusted servers only see the encrypted data record values and never see the actual data record values. However, this method of preserving data privacy is not secure. For example, it allows other parties to deduce whether two data record values are the same.

[0004] Concerns over data control and protection may be mitigated if a probabilistic encryption scheme is applied to the data record values before they are stored in the un-trusted servers. In doing so, multiple encryptions of one data record value can produce different encrypted values. However, the obvious challenge would then be how to match probabilistic encryption data when that very one data record can produce different encrypted values?

[0005] The object of the invention is thus to overcome the above problems and provide a new method for matching probabilistic encryption data.

SUMMARY OF INVENTION

[0006] According to a first aspect of the invention, a method for determining whether a first encrypted data of a first data value is equal to a second encrypted data of a second data value is described, the method comprising the steps of composing a first cyclic group, the first cyclic group comprising a plurality of elements; and composing a second cyclic group, the

second cyclic group comprising a plurality of elements including a first element. The method further comprises the step of applying a mathematical operation to the first cyclic group to map elements of the first cyclic group to one of the elements in the second cyclic group. The method further comprises the steps of randomly selecting a second element from the first cyclic group; and producing the first encrypted data by mapping the second element and the first data value into one or more elements of the first cyclic group. The method further comprises the steps of randomly selecting a third element from the first cyclic group; and producing the second encrypted data by mapping the third element and the second data value into one or more elements of the first cyclic group. The method further comprises the step of performing a test condition by applying the mathematical operation to the first encrypted data and the second encrypted data to obtain a fourth element in the second cyclic group, wherein the fourth element is equal to the first element when the first data value is equal to the second data value.

[0007] Preferably, the method further comprises the steps of randomly selecting integers to form a secret key; and generating a token, the token being a function of the secret key. Wherein the step of producing the first encrypted data comprises the step of mapping the second element, the first data value and the secret key into one or more elements of the first cyclic group. Wherein the step of producing the second encrypted data comprises the step of mapping the third element, the second data value and the secret key into one or more elements of the first cyclic group. Wherein the step of performing a test condition comprises the step of applying the mathematical operation to the first encrypted data, the second encrypted data and the token.

[0008] Preferably, the mathematical operation is a bilinear mapping operation.

[0009] Preferably, the first element is an identity element of the second cyclic group.

[0010] According to a second aspect of the invention, a method for determining which probabilistically encrypted values in a first set is equal to the probabilistically encrypted values in a second set is described, the method comprising the steps of extracting a first data value from the first set; and extracting a second data value from the second set. The method further comprises the step of determining whether a first encrypted data of the first data value is equal to a second encrypted data of the second data value by using the method as described in the first aspect of the invention.

[0011] According to a third aspect of the invention, a method for determining which probabilistically encrypted values in a first table is equal to the probabilistically encrypted values in a second table is described, the method comprising the steps of extracting a first record from the first table, the first record having a first attribute with a first data value; and extracting a second record from the second table, the second record having a second attribute with a second data value. The method further comprises the step of determining whether a first encrypted data of the first data value is equal to a second encrypted data of the second data value by using the method as described in the first aspect of the invention.

[0012] According to a fourth aspect of the invention, a system for determining whether a first encrypted data of a first data value is equal to a second encrypted data of a second data value is described, the system comprising a client machine. The client machine is configured to compose a first cyclic group, the first cyclic group comprising a plurality of elements; and compose a second cyclic group, the second cyclic group comprising a plurality of elements including a first element. The client machine is further configured to apply a mathematical operation to the first cyclic group to map elements of the first cyclic group to one of the elements in the second cyclic group; randomly select a second element from the first cyclic group; and produce the first encrypted data by mapping the second element and the first data value into one or more elements of the first cyclic group. The client machine is further configured to randomly select a third element from the first cyclic group; and produce the second encrypted data by mapping the third element and the second data value into one or more elements of the first cyclic group. The system further comprises a server, the server configured to receive the first encrypted data and the second encrypted data from the client machine; and perform a test condition by applying the mathematical operation to the first encrypted data and the second encrypted data to obtain a fourth element in the second cyclic group, wherein the fourth element is equal to the first element when the first data value is equal to the second data value.

[0013] Preferably, the client machine is further configured to randomly select integers to form a secret key; and generate a token, the token being a function of the secret key. The client machine is further configured to produce the first encrypted data by mapping the second element, the first data value and the secret key into one or more elements of the first cyclic group; and produce the second encrypted data by mapping the third element, the second data value and the secret key into one or more elements of the first cyclic group.

[0014] Preferably, the server is further configured to receive the token from the client machine; and apply the mathematical operation to the first encrypted data, the second encrypted data and the token to obtain a fourth element in the second cyclic group, wherein the fourth element is equal to the first element when the first data value is equal to the second data value.

[0015] Preferably, the mathematical operation is a bilinear mapping operation.

[0016] Preferably, the first element is an identity element of the second cyclic group.

[0017] The invention will now be described in detail with reference to the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0018] The accompanying figures illustrate disclosed embodiment(s) and serve to explain principles of the disclosed embodiment(s). It is to be understood, however, that these drawings are presented for purposes of illustration only, and not for defining limits of the application.

[0019] Figure 1 is a flow chart that depicts a method for determining whether two probabilistically encrypted values are equal in accordance with a preferred embodiment of the invention.

[0020] Figure 2 is a flow chart that depicts a method for determining which probabilistically encrypted values in a first set of values match with the probabilistically encrypted values in a second set of values in accordance with a preferred embodiment of the invention.

[0021] Figure 3 is a flow chart that depicts a equijoin method to discover pairs of records from two tables, whose attribute value in a record in the first table matches with an attribute value in a record in the second table.

[0022] Figure 4 depicts a system for implementing the method in accordance with a preferred embodiment of the invention.

[0023] Exemplary, non-limiting embodiments of the present application will now be described with references to the above-mentioned figures.

DETAILED DESCRIPTION

[0024] Figure 1 shows a method for determining whether two probabilistically encrypted values are equal.

[0025] Referring to step 101 in figure 1, two cyclic groups G and G_T with bilinear mapping e: G x G $\rightarrow$ G_T are composed. G is the first cyclic group and G_T is the second cyclic group. The bilinear mapping operation is applied to the first cyclic group G, so as to map elements of the first cyclic group G to one of the elements in the second cyclic group G_T. The group structure G has prime order p. The group structure G has a plurality of elements, among which is a generator g_1. As g_1 is the generator, therefore $G_1 = \{g_1, g_1^2, g_1^3, \ldots, g_1^p\}$ and $g_1^p = 1$, where 1 is the identity element of G. The group structure G_T also has a plurality of elements and 1 is also the identity element of G_T. For any integers a and b, $e(g_1^a, g_1^b) = e(g_1, g_1)^{ab}$.

[0026] In step 102, a client machine chooses random integers σ, σ_1 and σ_2 in the range [1..p] and computes $g_2 = g_1^{\sigma}$, where g_2 is an element of G. g_2 is used to give a safeguarded form of secret value σ to the server. The server needs g_2 to compute the test condition. The server cannot get the actual value of σ without doing a discrete log operation, which is a hard computational problem.

[0027] In step 103, the client machine stores σ, σ_1 and σ_2 as a secret key, and releases p, g_1 and g_2 to the un-trusted server as a public key.

[0028] In step 104, the client machine randomly selects integers λ_1, λ_2, κ_A and τ_A in the range [1...p], and randomly selects element x from the first cyclic group G. The client machine adds κ_A and τ_A to the secret key.

[0029] In step 105, for a first data value u, the client machine uses the secret key to generate encrypted data A having eight components i.e. A = (A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8), where $A_1 = x^{\lambda_1}$, $A_2 = g_1^{\lambda_2}$, $A_3 = x^{\sigma_1 \times u + \sigma_2} \cdot g_2^{\lambda_1 + \lambda_2}$, $A_4 = x$, $A_5 = x^{\sigma}$, $A_6 = x^{\sigma/\kappa_A}$, $A_7 = g_1^{\lambda_1 \times \tau_A}$,

$A_8 = e(x,x)^{\sigma_1 \times u + \sigma_2}$. One skilled in the art will appreciate that encrypted data A does not necessarily need eight components and the eight components described here is used as an illustration for the preferred embodiment.

[0030] In step 106, the client machine randomly selects integers μ_1, μ_2, κ_B and τ_B in the range [1...p], and randomly selects element y from the first cyclic group G. The client machine adds κ_B and τ_B to the secret key.

[0031] In step 107, for a second data value v, the client machine uses the secret key to generate encrypted data B having eight components i.e. B = (B_1, B_2, B_3, B_4, B_5, B_6, B_7, B_8), where $B_1 = y^{\mu_1}$, $B_2 = g_1^{\mu_2}$, $B_3 = y^{\sigma_1 \times v + \sigma_2} \cdot g_2^{\mu_1 + \mu_2}$, $B_4 = y$, $B_5 = y^{\sigma}$, $B_6 = y^{\sigma/\kappa_B}$, $B_7 = g_1^{\mu_1 \times \tau_B}$, $B_8 = e(y,y)^{\sigma_1 \times v + \sigma_2}$. One skilled in the art will appreciate that encrypted data B does not necessarily need eight components and the eight components described here is used as an illustration for the preferred embodiment.

[0032] In step 108, the client machine then deposits encrypted data A and encrypted data B in un-trusted server.

[0033] To test whether first data value u and second data value v are equal, in step 109, the client machine generates a token $\tau_{AB} = (\kappa_B/\tau_A, -\kappa_A/\tau_B)$, and sends token τ_{AB} to the un-trusted server. The token is a function of the secret key (as the secret key comprises of $\kappa_A, \kappa_B, \tau_A, \tau_B$).

[0034] In step 110, the un-trusted server then applies bilinear mappings to components of encrypted data A and encrypted data B to result in the test condition $\frac{e(A_3 \cdot B_3^{-1}, A_4 \cdot B_4)}{A_8 \cdot B_8^{-1} \cdot e(A_1 \cdot B_1^{-1}, g_2) \cdot e(A_7, B_6^{\kappa_B/\tau_A}) \cdot e(B_7^{-\kappa_A/\tau_B}, A_6) \cdot e(A_2 \cdot B_2^{-1}, A_5 \cdot B_5)} = 1$ to determine whether first data value u and second data value v are equal. $\frac{e(A_3 \cdot B_3^{-1}, A_4 \cdot B_4)}{A_8 \cdot B_8^{-1} \cdot e(A_1 \cdot B_1^{-1}, g_2) \cdot e(A_7, B_6^{\kappa_B/\tau_A}) \cdot e(B_7^{-\kappa_A/\tau_B}, A_6) \cdot e(A_2 \cdot B_2^{-1}, A_5 \cdot B_5)}$ is an element in G_T and 1 is the identity element of G_T. The test condition involves applying bilinear mappings to components of encrypted data A, components of encrypted data B, token τ_{AB} and g_2.

[0035] If $\frac{e(A_3 \cdot B_3^{-1}, A_4 \cdot B_4)}{A_8 \cdot B_8^{-1} \cdot e(A_1 \cdot B_1^{-1}, g_2) \cdot e(A_7, B_6^{\kappa_B/\tau_A}) \cdot e(B_7^{-\kappa_A/\tau_B}, A_6) \cdot e(A_2 \cdot B_2^{-1}, A_5 \cdot B_5)} = 1$, then first data value u and second data value v are equal.

[0036] The components of the denominator of the test condition works out to be:-

$$A_8 \cdot B_8^{-1} = e(x,x)^{\sigma_1 \times u + \sigma_2} . e(y,y)^{-\sigma_1 \times v - \sigma_2}$$

$$e(A_1 \cdot B_1^{-1}, g_2) = e\left(x^{\lambda_1} \cdot (y^{\mu_1})^{-1}, g_2\right) = e\left(x^{\lambda_1} \cdot y^{-\mu_1}, g_2\right) = e\left(g_2, x^{\lambda_1} \cdot y^{-\mu_1}\right)$$
$$= e\left(g_2, x^{\lambda_1}\right) \cdot e\left(g_2, y^{-\mu_1}\right) = e(g_2,x)^{\lambda_1} \cdot e(g_2,y)^{-\mu_1}$$

$$e\left(A_7, B_6^{\kappa_B/\tau_A}\right) = e\left(g_1^{\lambda_1 \times \tau_A}, (y^{\sigma/\kappa_B})^{\kappa_B/\tau_A}\right) = e\left(g_1^{\lambda_1 \times \tau_A}, y^{\sigma/\tau_A}\right) = e(g_1,y)^{\lambda_1 \times \tau_A \times \sigma/\tau_A}$$
$$= e(g_1,y)^{\lambda_1 \times \sigma} = e(g_1^{\sigma}, y)^{\lambda_1} = e(g_2,y)^{\lambda_1}$$

$$e\left(B_7^{-\kappa_A/\tau_B}, A_6\right) = e\left((g_1^{\mu_1 \times \tau_B})^{-\kappa_A/\tau_B}, x^{\sigma/\kappa_A}\right) = e\left(g_1^{-\mu_1 \times \kappa_A}, x^{\sigma/\kappa_A}\right) = e(g_1,x)^{-\mu_1 \times \kappa_A \times \sigma/\kappa_A}$$
$$= e(g_1,x)^{-\mu_1 \times \sigma} = e(g_1^{\sigma}, x)^{-\mu_1} = e(g_2,x)^{-\mu_1}$$

$$e(A_2 \cdot B_2^{-1}, A_5 \cdot B_5) = e\left(g_1^{\lambda_2} \cdot g_1^{-\mu_2}, x^{\sigma} \cdot y^{\sigma}\right) = e\left(g_1^{\lambda_2 - \mu_2}, (xy)^{\sigma}\right) = e(g_1, xy)^{(\lambda_2 - \mu_2)\sigma}$$
$$= e(g_1^{\sigma}, xy)^{\lambda_2 - \mu_2} = e(g_2, xy)^{\lambda_2 - \mu_2} = e(g_2,x)^{\lambda_2 - \mu_2} \cdot e(g_2,y)^{\lambda_2 - \mu_2}$$

[0037] Therefore, the denominator of the test condition works out to be:-

$$e(x,x)^{\sigma_1 \times u + \sigma_2} \cdot e(y,y)^{-\sigma_1 \times v - \sigma_2} \cdot e(g_2,x)^{\lambda_1} \cdot e(g_2,y)^{-\mu_1} \cdot e(g_2,y)^{\lambda_1} \cdot e(g_2,x)^{-\mu_1}$$
$$\cdot e(g_2,x)^{\lambda_2 - \mu_2} \cdot e(g_2,y)^{\lambda_2 - \mu_2}$$
$$= e(x,x)^{\sigma_1 \times u + \sigma_2} \cdot e(y,y)^{-\sigma_1 \times v - \sigma_2} \cdot e(g_2,x)^{\lambda_1 - \mu_1 + \lambda_2 - \mu_2} \cdot e(g_2,y)^{-\mu_1 + \lambda_1 + \lambda_2 - \mu_2}$$
$$= e(x,x)^{\sigma_1 \times u + \sigma_2} \cdot e(y,y)^{-\sigma_1 \times v - \sigma_2} \cdot e(g_2,x)^{\lambda_1 + \lambda_2 - \mu_1 - \mu_2} \cdot e(g_2,y)^{\lambda_1 + \lambda_2 - \mu_1 - \mu_2}$$

[0038] The numerator of the test condition works out to be:-

$$e(A_3 \cdot B_3^{-1}, A_4 \cdot B_4) = e\left(x^{\sigma_1 \times u + \sigma_2} \cdot g_2^{\lambda_1 + \lambda_2} \cdot y^{-\sigma_1 \times v - \sigma_2} \cdot g_2^{-\mu_1 - \mu_2}, x \cdot y\right)$$
$$= e\left(x^{\sigma_1 \times u + \sigma_2} \cdot y^{-\sigma_1 \times v - \sigma_2} \cdot g_2^{\lambda_1 + \lambda_2 - \mu_1 - \mu_2}, x \cdot y\right)$$
$$= e\left(x^{\sigma_1 \times u + \sigma_2} \cdot y^{-\sigma_1 \times v - \sigma_2}, x \cdot y\right) \cdot e\left(g_2^{\lambda_1 + \lambda_2 - \mu_1 - \mu_2}, x \cdot y\right)$$
$$= e\left(x^{\sigma_1 \times u + \sigma_2}, x\right) \cdot e\left(y^{-\sigma_1 \times v - \sigma_2}, x\right) \cdot e\left(x^{\sigma_1 \times u + \sigma_2}, y\right) \cdot e\left(y^{-\sigma_1 \times v - \sigma_2}, y\right) \cdot e\left(g_2^{\lambda_1 + \lambda_2 - \mu_1 - \mu_2}, x \cdot y\right)$$
$$= e(x,x)^{\sigma_1 \times u + \sigma_2} \cdot e(y,x)^{-\sigma_1 \times v - \sigma_2} \cdot e(x,y)^{\sigma_1 \times u + \sigma_2} \cdot e(y,y)^{-\sigma_1 \times v - \sigma_2} \cdot e\left(g_2^{\lambda_1 + \lambda_2 - \mu_1 - \mu_2}, x\right)$$
$$\cdot e\left(g_2^{\lambda_1 + \lambda_2 - \mu_1 - \mu_2}, y\right)$$
$$= e(x,x)^{\sigma_1 \times u + \sigma_2} \cdot e(x,y)^{-\sigma_1 \times v - \sigma_2} \cdot e(x,y)^{\sigma_1 \times u + \sigma_2} \cdot e(y,y)^{-\sigma_1 \times v - \sigma_2} \cdot e(g_2,x)^{\lambda_1 + \lambda_2 - \mu_1 - \mu_2}$$
$$\cdot e(g_2,y)^{\lambda_1 + \lambda_2 - \mu_1 - \mu_2}$$

$$= e(x,x)^{\sigma_1 \times u + \sigma_2} \cdot e(x,y)^{\sigma_1(u-v)} \cdot e(y,y)^{-\sigma_1 \times v - \sigma_2} \cdot e(g_2,x)^{\lambda_1+\lambda_2-\mu_1-\mu_2} \cdot e(g_2,y)^{\lambda_1+\lambda_2-\mu_1-\mu_2}$$

[0039] Therefore, the test condition works out to be:-

$$\frac{e(x,x)^{\sigma_1 \times u + \sigma_2} \cdot e(x,y)^{\sigma_1(u-v)} \cdot e(y,y)^{-\sigma_1 \times v - \sigma_2} \cdot e(g_2,x)^{\lambda_1+\lambda_2-\mu_1-\mu_2} \cdot e(g_2,y)^{\lambda_1+\lambda_2-\mu_1-\mu_2}}{e(x,x)^{\sigma_1 \times u + \sigma_2} \cdot e(y,y)^{-\sigma_1 \times v - \sigma_2} \cdot e(g_2,x)^{\lambda_1+\lambda_2-\mu_1-\mu_2} \cdot e(g_2,y)^{\lambda_1+\lambda_2-\mu_1-\mu_2}} = e(x,y)^{\sigma_1(u-v)}.$$

[0040] When first data value u and second data value v are equal, $e(x,y)^{\sigma_1(u-v)} = e(x,y)^{\sigma_1(0)} = e(x,y)^0 = 1$, where 1 is the identity element of G_T. Therefore, if test condition $\frac{e(A_3 \cdot B_3^{-1}, A_4 \cdot B_4)}{A_8 \cdot B_8^{-1} \cdot e(A_1 \cdot B_1^{-1}, g_2) \cdot e(A_7, B_6^{\kappa_B/\tau_A}) \cdot e(B_7^{-\kappa_A/\tau_B}, A_6) \cdot e(A_2 \cdot B_2^{-1}, A_5 \cdot B_5)} = 1$ holds, un-trusted server determines that first data value u is equal to second data value v.

[0041] This encryption method is advantageous as it is probabilistic. This is because integers λ_1 and λ_2 as well as element x are randomly selected each time first data value u is encrypted, and integers μ_1 and μ_2 as well as element y are randomly selected each time second data value v is encrypted. Therefore, for data value u or v, encrypted at different times, the encrypted data A and B will be different. Despite the fact that every encryption of first data value u and second data value v will generate different encrypted data A and B, the test condition $\frac{e(A_3 \cdot B_3^{-1}, A_4 \cdot B_4)}{A_8 \cdot B_8^{-1} \cdot e(A_1 \cdot B_1^{-1}, g_2) \cdot e(A_7, B_6^{\kappa_B/\tau_A}) \cdot e(B_7^{-\kappa_A/\tau_B}, A_6) \cdot e(A_2 \cdot B_2^{-1}, A_5 \cdot B_5)} = 1$ will always determine correctly whether first data value u is equal to second data value v. This is due to the specific structure of the test condition, encrypted data A and B and token τ_{AB}, which results in the condition $e(x,y)^{\sigma_1(u-v)} = 1$, so that integer σ_1, element x and element y will be neutralized when first data value u is equal to second data value v.

[0042] Another advantage of the encryption method is that the un-trusted server can perform the test condition only after receiving token τ_{AB} from the client machine.

[0043] Figure 2 shows a method for determining which probabilistically encrypted values in a first set of values $U = \{u_1, u_2, \cdots, u_m\}$, match with the probabilistically encrypted values in a second set of values $V = \{v_1, v_2, \cdots, v_m\}$.

[0044] Referring to step 201 in figure 2, two cyclic groups G and G_T with bilinear mapping e: G x G $\rightarrow$ G_T are composed. G is the first cyclic group and G_T is the second cyclic

group. The bilinear mapping operation is applied to the first cyclic group G, so as to map elements of the first cyclic group G to one of the elements in the second cyclic group G_T. The group structure G has prime order p. The group structure G has a plurality of elements, among which is a generator g_1. As g_1 is the generator, therefore $G_1 = \{g_1, g_1^2, g_1^3, \ldots, g_1^p\}$ and $g_1^p = 1$, where 1 is the identity element of G. The group structure G_T also has a plurality of elements and 1 is also the identity element of G_T. For any integers a and b, $e(g_1^a, g_1^b) = e(g_1, g_1)^{ab}$.

[0045] In step 202, a client machine chooses random integers σ, σ_1 and σ_2 in the range [1...p] and computes $g_2 = g_1^{\sigma}$, where g_2 is an element of G. g_2 is used to give a safeguarded form of secret value σ to the server. The server needs g_2 to compute the test condition. The server cannot get the actual value of σ without doing a discrete log operation, which is a hard computational problem.

[0046] In step 203, the client machine stores σ, σ_1 and σ_2 as a secret key, and releases p, g_1 and g_2 to the un-trusted server as a public key.

[0047] In step 204, for the first set of values $U = \{u_1, u_2, \cdots, u_m\}$, the client machine randomly selects integers κ_A and τ_A in the range [1...p]. The client machine adds κ_A and τ_A to the secret key.

[0048] In step 205, for every value u_i of U, the client machine randomly selects integers $\lambda_{i,1}$ and $\lambda_{i,2}$ in the range [1...p], and randomly selects element x_i from the first cyclic group G.

[0049] In step 206, for every value u_i of U, the client machine generates encrypted data $A_i = (A_{i,1}, A_{i,2}, A_{i,3}, A_{i,4}, A_{i,5}, A_{i,6}, A_{i,7}, A_{i,8})$, where $A_{i,1} = x_i^{\lambda_{i,1}}$, $A_{i,2} = g_1^{\lambda_{i,2}}$, $A_{i,3} = x_i^{\sigma_1 \times u_i + \sigma_2} \cdot g_2^{\lambda_{i,1} + \lambda_{i,2}}$, $A_{i,4} = x_i$, $A_{i,5} = x_i^{\sigma}$, $A_{i,6} = x_i^{\sigma/\kappa_A}$, $A_{i,7} = g_1^{\lambda_{i,1} \times \tau_A}$, $A_{i,8} = e(x_i, x_i)^{\sigma_1 \times u_i + \sigma_2}$. One skilled in the art will appreciate that encrypted data A_i does not necessarily need eight components and the eight components described here is used as an illustration for the preferred embodiment.

[0050] In step 207, for the second set of values $V = \{v_1, v_2, \cdots, v_n\}$, the client machine randomly selects integers κ_B and τ_B in the range [1...p]. The client machine adds κ_B and τ_B to the secret key.

[0051] In step 208, for every value v_j of V, the client machine randomly selects integers $\mu_{j,1}$ and $\mu_{j,2}$ in the range [1...p], and randomly selects element y_j from the first cyclic group G.

[0052] In step 209, for every value v_j of V, the client machine generates encrypted data $B_j = (B_{j,1}, B_{j,2}, B_{j,3}, B_{j,4}, B_{j,5}, B_{j,6}, B_{j,7}, B_{i,8})$, where $B_{j,1} = y_j^{\mu_{j,1}}$, $B_{j,2} = g_1^{\mu_{j,2}}$, $B_{j,3} = y_j^{\sigma_1 \times v_j + \sigma_2} \cdot g_2^{\mu_{j,1}+\mu_{j,2}}$, $B_{j,4} = y_j$, $B_{j,5} = y_j^{\sigma}$, $B_{j,6} = y_j^{\sigma/\kappa_B}$, $B_{j,7} = g_1^{\mu_{j,1} \times \tau_B}$, $B_{j,8} = e(y_j, y_j)^{\sigma_1 \times v_j + \sigma_2}$. One skilled in the art will appreciate that encrypted data B_j does not necessarily need eight components and the eight components described here is used as an illustration for the preferred embodiment.

[0053] In step 210, the client machine then deposits encrypted data A_i for U and encrypted data B_j for V in the un-trusted server.

[0054] To test whether u_i of U and v_j of V are equal, in step 211, the client machine generates a token $\tau_{AB} = (\kappa_B/\tau_A, -\kappa_A/\tau_B)$, and sends token τ_{AB} to the un-trusted server. The token is a function of the secret key (as the secret key comprises of $\kappa_A, \kappa_B, \tau_A, \tau_B$).

[0055] In step 212, the un-trusted server then applies bilinear mappings to components of encrypted data A_i and encrypted data B_j to result in the test condition $\frac{e(A_{i,3} \cdot B_{j,3}^{-1}, A_{i,4} \cdot B_{j,4})}{A_{i,8} \cdot B_{j,8}^{-1} \cdot e(A_{i,1} \cdot B_{j,1}^{-1}, g_2) \cdot e(A_{i,7}, B_{j,6}^{\kappa_B/\tau_A}) \cdot e(B_{j,7}^{-\kappa_A/\tau_B}, A_{i,6}) \cdot e(A_{i,2} \cdot B_{j,2}^{-1}, A_{i,5} \cdot B_{j,5})} = 1$, for every u_i of U and v_j of V, to determine whether u_i of U matches v_j of V. $\frac{e(A_{i,3} \cdot B_{j,3}^{-1}, A_{i,4} \cdot B_{j,4})}{A_{i,8} \cdot B_{j,8}^{-1} \cdot e(A_{i,1} \cdot B_{j,1}^{-1}, g_2) \cdot e(A_{i,7}, B_{j,6}^{\kappa_B/\tau_A}) \cdot e(B_{j,7}^{-\kappa_A/\tau_B}, A_{i,6}) \cdot e(A_{i,2} \cdot B_{j,2}^{-1}, A_{i,5} \cdot B_{j,5})}$ is an element in G_T and 1 is the identity element of G_T. The test condition involves applying bilinear mappings to components of encrypted data A_i, components of encrypted data B_j, token τ_{AB} and g_2.

[0056] If $\frac{e(A_{i,3} \cdot B_{j,3}^{-1}, A_{i,4} \cdot B_{j,4})}{A_{i,8} \cdot B_{j,8}^{-1} \cdot e(A_{i,1} \cdot B_{j,1}^{-1}, g_2) \cdot e(A_{i,7}, B_{j,6}^{\kappa_B/\tau_A}) \cdot e(B_{j,7}^{-\kappa_A/\tau_B}, A_{i,6}) \cdot e(A_{i,2} \cdot B_{j,2}^{-1}, A_{i,5} \cdot B_{j,5})} = 1$, then u_i matches v_j.

[0057] The components of the denominator of the test condition works out to be:-

$$A_{i,8} \cdot B_{j,8}^{-1} = e(x_i, x_i)^{\sigma_1 \times u_i + \sigma_2} \cdot e(y_j, y_j)^{-\sigma_1 \times v_j - \sigma_2}$$

$$e(A_{i,1} \cdot B_{j,1}^{-1}, g_2) = e\left(x_i^{\lambda_{i,1}} \cdot \left(y_j^{\mu_{j,1}}\right)^{-1}, g_2\right) = e\left(x_i^{\lambda_{i,1}} \cdot y_j^{-\mu_{j,1}}, g_2\right) = e\left(g_2, x_i^{\lambda_{i,1}} \cdot y_j^{-\mu_{j,1}}\right)$$
$$= e\left(g_2, x_i^{\lambda_{i,1}}\right) \cdot e\left(g_2, y_j^{-\mu_{j,1}}\right) = e(g_2, x_i)^{\lambda_{i,1}} \cdot e(g_2, y_j)^{-\mu_{j,1}}$$

$$\left(A_{i,7}, B_{j,6}^{\kappa_B/\tau_A}\right) = e\left(g_1^{\lambda_{i,1} \times \tau_A}, (y_j^{\sigma/\kappa_B})^{\kappa_B/\tau_A}\right) = e\left(g_1^{\lambda_{i,1} \times \tau_A}, y_j^{\sigma/\tau_A}\right) = e(g_1, y_j)^{\lambda_{i,1} \times \tau_A \times \sigma/\tau_A}$$
$$= e(g_1, y_j)^{\lambda_{i,1} \times \sigma} = e(g_1^{\sigma}, y_j)^{\lambda_{i,1}} = e(g_2, y_j)^{\lambda_{i,1}}$$

$$e\left(B_{j,7}^{-\kappa_A/\tau_B}, A_{i,6}\right) = e\left((g_1^{\mu_{j,1} \times \tau_B})^{-\kappa_A/\tau_B}, x_i^{\sigma/\kappa_A}\right) = e\left(g_1^{-\mu_{j,1} \times \kappa_A}, x_i^{\sigma/\kappa_A}\right)$$
$$= e(g_1, x_i)^{-\mu_{j,1} \times \kappa_A \times \sigma/\kappa_A} = e(g_1, x_i)^{-\mu_{j,1} \times \sigma} = e(g_1^{\sigma}, x_i)^{-\mu_{j,1}} = e(g_2, x_i)^{-\mu_{j,1}}$$

$$e(A_{i,2} \cdot B_{j,2}^{-1}, A_{i,5} \cdot B_{j,5}) = e\left(g_1^{\lambda_{i,2}} \cdot g_1^{-\mu_{j,2}}, x_i^{\sigma} \cdot y_j^{\sigma}\right) = e\left(g_1^{\lambda_{i,2} - \mu_{j,2}}, (x_i y_j)^{\sigma}\right)$$
$$= e(g_1, x_i y_j)^{(\lambda_{i,2} - \mu_{j,2})\sigma}$$
$$= e(g_1^{\sigma}, x_i y_j)^{\lambda_{i,2} - \mu_{j,2}} = e(g_2, x_i y_j)^{\lambda_{i,2} - \mu_{j,2}}$$
$$= e(g_2, x_i)^{\lambda_{i,2} - \mu_{j,2}} \cdot e(g_2, y_j)^{\lambda_{i,2} - \mu_{j,2}}$$

[0058] Therefore, the denominator of the test condition works out to be

$$e(x_i, x_i)^{\sigma_1 \times u_i + \sigma_2} \cdot e(y_j, y_j)^{-\sigma_1 \times v_j - \sigma_2} \cdot e(g_2, x_i)^{\lambda_{i,1}} \cdot$$
$$e(g_2, y_j)^{-\mu_{j,1}} \cdot e(g_2, y_j)^{\lambda_{i,1}} \cdot e(g_2, x_i)^{-\mu_{j,1}} \cdot e(g_2, x_i)^{\lambda_{i,2} - \mu_{j,2}} \cdot e(g_2, y_j)^{\lambda_{i,2} - \mu_{j,2}}$$
$$= e(x_i, x_i)^{\sigma_1 \times u_i + \sigma_2} \cdot e(y_j, y_j)^{-\sigma_1 \times v_j - \sigma_2} \cdot e(g_2, x_i)^{\lambda_{i,1} - \mu_{j,1} + \lambda_{i,2} - \mu_{j,2}} \cdot e(g_2, y_j)^{-\mu_{j,1} + \lambda_{i,1} + \lambda_{i,2} - \mu_{j,2}}$$
$$= e(x_i, x_i)^{\sigma_1 \times u_i + \sigma_2} \cdot e(y_j, y_j)^{-\sigma_1 \times v_j - \sigma_2} \cdot e(g_2, x_i)^{\lambda_{i,1} + \lambda_{i,2} - \mu_{j,1} - \mu_{j,2}} \cdot e(g_2, y_j)^{\lambda_{i,1} + \lambda_{i,2} - \mu_{j,1} - \mu_{j,2}}$$

[0059] The numerator of the test condition works out to be:-

$$e(A_{i,3} \cdot B_{j,3}^{-1}, A_{i,4} \cdot B_{j,4}) = e\left(x_i^{\sigma_1 \times u_i + \sigma_2} \cdot g_2^{\lambda_{i,1} + \lambda_{i,2}} \cdot y_j^{-\sigma_1 \times v_j - \sigma_2} \cdot g_2^{-\mu_{j,1} - \mu_{j,2}}, x_i . y_j\right)$$
$$= e\left(x_i^{\sigma_1 \times u_i + \sigma_2} \cdot y_j^{-\sigma_1 \times v_j - \sigma_2} \cdot g_2^{\lambda_{i,1} + \lambda_{i,2} - \mu_{j,1} - \mu_{j,2}}, x_i . y_j\right)$$
$$= e\left(x_i^{\sigma_1 \times u_i + \sigma_2} \cdot y_j^{-\sigma_1 \times v_j - \sigma_2}, x_i . y_j\right) \cdot e\left(g_2^{\lambda_{i,1} + \lambda_{i,2} - \mu_{j,1} - \mu_{j,2}}, x_i . y_j\right)$$
$$= e\left(x_i^{\sigma_1 \times u_i + \sigma_2}, x_i\right) \cdot e\left(y_j^{-\sigma_1 \times v_j - \sigma_2}, x_i\right) \cdot e\left(x_i^{\sigma_1 \times u_i + \sigma_2}, y_j\right) \cdot e\left(y_j^{-\sigma_1 \times v_j - \sigma_2}, y_j\right)$$
$$\cdot e\left(g_2^{\lambda_{i,1} + \lambda_{i,2} - \mu_{j,1} - \mu_{j,2}}, x_i . y_j\right)$$
$$= e(x_i, x_i)^{\sigma_1 \times u_i + \sigma_2} \cdot e(y_j, x_i)^{-\sigma_1 \times v_j - \sigma_2} \cdot e(x_i, y_j)^{\sigma_1 \times u_i + \sigma_2} \cdot$$
$$e(y_j, y_j)^{-\sigma_1 \times v_j - \sigma_2} \cdot e\left(g_2^{\lambda_{i,1} + \lambda_{i,2} - \mu_{j,1} - \mu_{j,2}}, x_i\right) \cdot e\left(g_2^{\lambda_{i,1} + \lambda_{i,2} - \mu_{j,1} - \mu_{j,2}}, y_j\right)$$
$$= e(x_i, x_i)^{\sigma_1 \times u_i + \sigma_2} \cdot e(x_i, y_j)^{-\sigma_1 \times v_j - \sigma_2} \cdot e(x_i, y_j)^{\sigma_1 \times u_i + \sigma_2} \cdot e(y_j, y_j)^{-\sigma_1 \times v_j - \sigma_2} \cdot$$
$$e(g_2, x_i)^{\lambda_{i,1} + \lambda_{i,2} - \mu_{j,1} - \mu_{j,2}} \cdot e(g_2, y_j)^{\lambda_{i,1} + \lambda_{i,2} - \mu_{j,1} - \mu_{j,2}}$$

$$= e(x_i, x_i)^{\sigma_1 \times u_i + \sigma_2} \cdot e(x_i, y_j)^{\sigma_1 (u_i - v_j)} \cdot e(y_j, y_j)^{-\sigma_1 \times v_j - \sigma_2} \cdot e(g_2, x_i)^{\lambda_{i,1} + \lambda_{i,2} - \mu_{j,1} - \mu_{j,2}} \cdot e(g_2, y_j)^{\lambda_{i,1} + \lambda_{i,2} - \mu_{j,1} - \mu_{j,2}}$$

[0060] Therefore, the test condition works out to be:-

$$\frac{e(x_i, x_i)^{\sigma_1 \times u_i + \sigma_2} \cdot e(x_i, y_j)^{\sigma_1 (u_i - v_j)} \cdot e(y_j, y_j)^{-\sigma_1 \times v_j - \sigma_2} \cdot e(g_2, x_i)^{\lambda_{i,1} + \lambda_{i,2} - \mu_{j,1} - \mu_{j,2}} \cdot e(g_2, y_j)^{\lambda_{i,1} + \lambda_{i,2} - \mu_{j,}}}{e(x_i, x_i)^{\sigma_1 \times u_i + \sigma_2} \cdot e(y_j, y_j)^{-\sigma_1 \times v_j - \sigma_2} \cdot e(g_2, x_i)^{\lambda_{i,1} + \lambda_{i,2} - \mu_{j,1} - \mu_{j,2}} \cdot e(g_2, y_j)^{\lambda_{i,1} + \lambda_{i,2} - \mu_{j,1} - \mu_{j,2}}}$$

$$= e(x_i, y_j)^{\sigma_1 (u_i - v_j)}$$

[0061] When u_i is equal to v_j, then $e(x_i, y_j)^{\sigma_1 (u_i - v_j)} = e(x_i, y_j)^0 = 1$, where 1 is the identity element of G_T. Therefore, if test condition $\frac{e\left(A_{i,3} \cdot B_{j,3}^{-1}, A_{i,4} \cdot B_{j,4}\right)}{A_{i,8} \cdot B_{j,8}^{-1} \cdot e\left(A_{i,1} \cdot B_{j,1}^{-1}, g_2\right) \cdot e(A_{i,7}, B_{j,6}^{\kappa_B / \tau_A}) \cdot e(B_{j,7}^{-\kappa_A / \tau_B}, A_{i,6}) \cdot e(A_{i,2} \cdot B_{j,2}^{-1}, A_{i,5} \cdot B_{j,5})} = 1$ holds, un-trusted server will determine that u_i matches v_j.

[0062] This encryption method is advantageous as it is probabilistic. This is because integers $\lambda_{i,1}$ and $\lambda_{i,2}$ as well as element x_i are randomly selected each time u_i is encrypted, and integers $\mu_{j,1}$ and $\mu_{j,2}$ as well as element y_j are randomly selected each time v_j is encrypted. Therefore, for u_i and v_j, encrypted at different times, the encrypted data A_i and B_j will be different. Despite the fact that every encryption of u_i and v_j will result in different encrypted data A_i and B_j, the test condition $\frac{e\left(A_{i,3} \cdot B_{j,3}^{-1}, A_{i,4} \cdot B_{j,4}\right)}{A_{i,8} \cdot B_{j,8}^{-1} \cdot e\left(A_{i,1} \cdot B_{j,1}^{-1}, g_2\right) \cdot e(A_{i,7}, B_{j,6}^{\kappa_B / \tau_A}) \cdot e(B_{j,7}^{-\kappa_A / \tau_B}, A_{i,6}) \cdot e(A_{i,2} \cdot B_{j,2}^{-1}, A_{i,5} \cdot B_{j,5})} = 1$ will always determine correctly whether u_i is equal to v_j. This is due to the specific structure of the test condition, encrypted data A_i and B_j and token τ_{AB}, which results in the condition $e(x_i, y_j)^{\sigma_1 (u_i - v_j)} = 1$, so that integer σ_1, element x_i and element y_j will be neutralized when u_i is equal to v_j.

[0063] Another advantage of the encryption method is that the un-trusted server can perform the test condition only after receiving token τ_{AB} from the client machine. Further still, token τ_{AB} can only be used for discovering matching values across U and V specifically, and cannot be used for other sets of values.

[0064] Figure 3 shows a method for performing an equality join (or equijoin) on two tables. An equijoin is one of the most common operations in a relational Database

Management System. An equijoin on two tables is able to determine if an attribute in a record in the first table is equal in value to an attribute in a record in the second table. Assume first table $R = \{r_1, r_2, \cdots, r_m\}$ and second table $S = \{s_1, s_2, \cdots, s_n\}$. Each record in R (r_i) has attribute u with attribute value $r_i.u$. Each record in S (s_j) has attribute v with attribute value $s_j.v$. The equijoin method involves discovering pairs of R and S records, whose attribute value u in a record in R matches with attribute value v in a record in S.

[0065] Referring to step 301 in figure 3, two cyclic groups G and G_T with bilinear mapping e: G x G $\rightarrow$ G_T are composed. G is the first cyclic group and G_T is the second cyclic group. The bilinear mapping operation is applied to the first cyclic group G, so as to map elements of the first cyclic group G to one of the elements in the second cyclic group G_T. The group structure G has prime order p. The group structure G has a plurality of elements, among which is a generator g_1. As g_1 is the generator, therefore $G_1 = \{g_1, g_1^2, g_1^3, \ldots, g_1^p\}$ and $g_1^p = 1$, where 1 is the identity element of G. The group structure G_T also has a plurality of elements and 1 is also the identity element of G_T. For any natural numbers a and b, $e(g_1^a, g_1^b) = e(g_1, g_1)^{ab}$.

[0066] In step 302, a client machine chooses random integers σ, σ_1 and σ_2 in the range [1...p] and computes $g_2 = g_1^{\sigma}$, where g_2 is an element of G. g_2 is used to give a safeguarded form of secret value σ to the server. The server needs g_2 to compute the test condition. The server cannot get the actual value of σ without doing a discrete log operation, which is a hard computational problem.

[0067] In step 303, the client machine stores σ, σ_1 and σ_2 as a secret key, and releases p, g_1 and g_2 to the un-trusted server as a public key.

[0068] In step 304, for the first table of records $R = \{r_1, r_2, \cdots, r_m\}$, the client machine randomly selects integers κ_A and τ_A in the range [1...p]. The client machine adds κ_A and τ_A to the secret key.

[0069] In step 305, for attribute value $r_i.u$ in each record r_i of R, the client machine randomly selects integers $\lambda_{i,1}$ and $\lambda_{i,2}$ in the range [1...p], and randomly selects element x_i from the first cyclic group G.

[0070] In step 306, for attribute value $r_i.u$ in each record r_i of R, the client machine generates encrypted data $A_i = (A_{i,1}, A_{i,2}, A_{i,3}, A_{i,4}, A_{i,5}, A_{i,6}, A_{i,7}, A_{i,8})$, where $A_{i,1} = x_i^{\lambda_{i,1}}$, $A_{i,2} = g_1^{\lambda_{i,2}}$, $A_{i,3} = x_i^{\sigma_1 \times r_i.u + \sigma_2} \cdot g_2^{\lambda_{i,1}+\lambda_{i,2}}$, $A_{i,4} = x_i$, $A_{i,5} = x_i^{\sigma}$, $A_{i,6} = x_i^{\sigma/\kappa_A}$, $A_{i,7} = g_1^{\lambda_{i,1} \times \tau_A}$, $A_{i,8} = e(x_i, x_i)^{\sigma_1 \times r_i.u + \sigma_2}$. One skilled in the art will appreciate that encrypted data A_i does not necessarily need eight components and the eight components described here is used as an illustration for the preferred embodiment.

[0071] In step 307, for the second table of records $S = \{s_1, s_2, \cdots, s_n\}$, the client machine randomly selects integers κ_B and τ_B in the range [1...p]. The client machine adds κ_B and τ_B to the secret key.

[0072] In step 308, for attribute value $s_j.v$ in each record s_j of S, the client machine randomly selects integers $\mu_{j,1}$ and $\mu_{j,2}$ in the range [1...p], and randomly selects element y_j from the first cyclic group G.

[0073] In step 309, for attribute value $s_j.v$ in each record s_j of S, the client machine generates encrypted data $B_j = (B_{j,1}, B_{j,2}, B_{j,3}, B_{j,4}, B_{j,5}, B_{j,6}, B_{j,7}, B_{j,8})$, where $B_{j,1} = y_j^{\mu_{j,1}}$, $B_{j,2} = g_1^{\mu_{j,2}}$, $B_{j,3} = y_j^{\sigma_1 \times s_j.v + \sigma_2} \cdot g_2^{\mu_{j,1}+\mu_{j,2}}$, $B_{j,4} = y_j$, $B_{j,5} = y_j^{\sigma}$, $B_{j,6} = y_j^{\sigma/\kappa_B}$, $B_{j,7} = g_1^{\mu_{j,1} \times \tau_B}$, $B_{j,8} = e(y_j, y_j)^{\sigma_1 \times s_j.v + \sigma_2}$. One skilled in the art will appreciate that encrypted data B_j does not necessarily need eight components and the eight components described here is used as an illustration for the preferred embodiment.

[0074] In step 310, the client machine then deposits the encrypted data A_i for each record r_i of R and encrypted data B_j for each record s_j of S in the un-trusted server.

[0075] To perform an equijoin of R and S on their attributes u and v, in step 311, the client machine generates a token $\tau_{AB} = (\kappa_B/\tau_A, -\kappa_A/\tau_B)$, and sends token τ_{AB} to the un-trusted server. The token is a function of the secret key (as the secret key comprises of $\kappa_A, \kappa_B, \tau_A, \tau_B$).

[0076] In step 312, the un-trusted server then applies bilinear mappings to components of encrypted data A_i and encrypted data B_j to result in the test condition

$\frac{e\left(A_{i,3}\cdot B_{j,3}^{-1},A_{i,4}\cdot B_{j,4}\right)}{A_{i,8}\cdot B_{j,8}^{-1}\cdot e\left(A_{i,1}\cdot B_{j,1}^{-1},g_2\right)\cdot e(A_{i,7},B_{j,6}^{\kappa_B/\tau_A})\cdot e(B_{j,7}^{-\kappa_A/\tau_B},A_{i,6})\cdot e(A_{i,2}\cdot B_{j,2}^{-1},A_{i,5}\cdot B_{j,5})} = 1$ for each record r_i of R and each record s_j of S, to determine whether attribute value $r_i.u$ matches attribute value $s_j.v$. $\frac{e\left(A_{i,3}\cdot B_{j,3}^{-1},A_{i,4}\cdot B_{j,4}\right)}{A_{i,8}\cdot B_{j,8}^{-1}\cdot e\left(A_{i,1}\cdot B_{j,1}^{-1},g_2\right)\cdot e(A_{i,7},B_{j,6}^{\kappa_B/\tau_A})\cdot e(B_{j,7}^{-\kappa_A/\tau_B},A_{i,6})\cdot e(A_{i,2}\cdot B_{j,2}^{-1},A_{i,5}\cdot B_{j,5})}$ is an element in G_T and 1 is the identity element of G_T. The test condition involves applying bilinear mappings to the components of encrypted data A_i, the components of encrypted data B_j, token τ_{AB} and g_2.

[0077] If $e\frac{e\left(A_{i,3}\cdot B_{j,3}^{-1},A_{i,4}\cdot B_{j,4}\right)}{A_{i,8}\cdot B_{j,8}^{-1}\cdot e\left(A_{i,1}\cdot B_{j,1}^{-1},g_2\right)\cdot e(A_{i,7},B_{j,6}^{\kappa_B/\tau_A})\cdot e(B_{j,7}^{-\kappa_A/\tau_B},A_{i,6})\cdot e(A_{i,2}\cdot B_{j,2}^{-1},A_{i,5}\cdot B_{j,5})} = 1,$

then attribute value $r_i.u$ matches attribute value $s_j.v$.

[0078] The components of the denominator of the test condition works out to be:-

$$A_{i,8}\cdot B_{j,8}^{-1} = e(x_i,x_i)^{\sigma_1\times u_i+\sigma_2}\cdot e(y_j,y_j)^{-\sigma_1\times v_j-\sigma_2}$$

$$e\left(A_{i,1}\cdot B_{j,1}^{-1},g_2\right) = e\left(x_i^{\lambda_{i,1}}\cdot\left(y_j^{\mu_{j,1}}\right)^{-1},g_2\right) = e\left(x_i^{\lambda_{i,1}}\cdot y_j^{-\mu_{j,1}},g_2\right) = e\left(g_2,x_i^{\lambda_{i,1}}\cdot y_j^{-\mu_{j,1}}\right) =$$
$$e\left(g_2,x_i^{\lambda_{i,1}}\right)\cdot e\left(g_2,y_j^{-\mu_{j,1}}\right) = e(g_2,x_i)^{\lambda_{i,1}}\cdot e(g_2,y_j)^{-\mu_{j,1}}$$

$$e\left(A_{i,7},B_{j,6}^{\kappa_B/\tau_A}\right) = e\left(g_1^{\lambda_{i,1}\times\tau_A},(y_j^{\sigma/\kappa_B})^{\kappa_B/\tau_A}\right) = e\left(g_1^{\lambda_{i,1}\times\tau_A},y_j^{\sigma/\tau_A}\right) = e(g_1,y_j)^{\lambda_{i,1}\times\tau_A\times\sigma/\tau_A} =$$
$$e(g_1,y_j)^{\lambda_{i,1}\times\sigma} = e(g_1^{\sigma},y_j)^{\lambda_{i,1}} = e(g_2,y_j)^{\lambda_{i,1}}$$

$$e\left(B_{j,7}^{-\kappa_A/\tau_B},A_{i,6}\right) = e\left((g_1^{\mu_{j,1}\times\tau_B})^{-\kappa_A/\tau_B},x_i^{\sigma/\kappa_A}\right) = e\left(g_1^{-\mu_{j,1}\times\kappa_A},x_i^{\sigma/\kappa_A}\right) =$$
$$e(g_1,x_i)^{-\mu_{j,1}\times\kappa_A\times\sigma/\kappa_A} = e(g_1,x_i)^{-\mu_{j,1}\times\sigma} = e(g_1^{\sigma},x_i)^{-\mu_{j,1}} = e(g_2,x_i)^{-\mu_{j,1}}$$

$$e\left(A_{i,2}\cdot B_{j,2}^{-1},A_{i,5}\cdot B_{j,5}\right) = e\left(g_1^{\lambda_{i,2}}\cdot g_1^{-\mu_{j,2}},x_i^{\sigma}\cdot y_j^{\sigma}\right) = e\left(g_1^{\lambda_{i,2}-\mu_{j,2}},(x_iy_j)^{\sigma}\right)$$
$$= e\left(g_1,x_iy_j\right)^{(\lambda_{i,2}-\mu_{j,2})\sigma}$$
$$= e(g_1^{\sigma},x_iy_j)^{\lambda_{i,2}-\mu_{j,2}} = e(g_2,x_iy_j)^{\lambda_{i,2}-\mu_{j,2}}$$
$$= e(g_2,x_i)^{\lambda_{i,2}-\mu_{j,2}}\cdot e(g_2,y_j)^{\lambda_{i,2}-\mu_{j,2}}$$

[0079] Therefore, the denominator of the test condition works out to be:-

$$e(x_i,x_i)^{\sigma_1\times r_i.u+\sigma_2}\cdot e(y_j,y_j)^{-\sigma_1\times s_j.v-\sigma_2}\cdot e(g_2,x_i)^{\lambda_{i,1}}\cdot$$
$$e(g_2,y_j)^{-\mu_{j,1}}.e(g_2,y_j)^{\lambda_{i,1}}.e(g_2,x_i)^{-\mu_{j,1}}.e(g_2,x_i)^{\lambda_{i,2}-\mu_{j,2}}\cdot e(g_2,y_j)^{\lambda_{i,2}-\mu_{j,2}}$$
$$= e(x_i,x_i)^{\sigma_1\times r_i.u+\sigma_2}\cdot e(y_j,y_j)^{-\sigma_1\times s_j.v-\sigma_2}\cdot e(g_2,x_i)^{\lambda_{i,1}-\mu_{j,1}+\lambda_{i,2}-\mu_{j,2}}\cdot$$
$$e(g_2,y_j)^{-\mu_{j,1}+\lambda_{i,1}+\lambda_{i,2}-\mu_{j,2}}$$

$$= e(x_i, x_i)^{\sigma_1 \times r_i.u + \sigma_2} \cdot e(y_j, y_j)^{-\sigma_1 \times s_j.v - \sigma_2} \cdot e(g_2, x_i)^{\lambda_{i,1}+\lambda_{i,2}-\mu_{j,1}-\mu_{j,2}} \cdot e(g_2, y_j)^{\lambda_{i,1}+\lambda_{i,2}-\mu_{j,1}-\mu_{j,2}}$$

[0080] The numerator of the test condition works out to be:-

$$e(A_{i,3} \cdot B_{j,3}^{-1}, A_{i,4} \cdot B_{j,4}) = e\left(x_i^{\sigma_1 \times r_i.u + \sigma_2} \cdot g_2^{\lambda_{i,1}+\lambda_{i,2}} \cdot y_j^{-\sigma_1 \times s_j.v - \sigma_2} \cdot g_2^{-\mu_{j,1}-\mu_{j,2}}, x_i.y_j\right)$$
$$= e\left(x_i^{\sigma_1 \times r_i.u + \sigma_2} \cdot y_j^{-\sigma_1 \times s_j.v - \sigma_2} \cdot g_2^{\lambda_{i,1}+\lambda_{i,2}-\mu_{j,1}-\mu_{j,2}}, x_i.y_j\right)$$
$$= e\left(x_i^{\sigma_1 \times r_i.u + \sigma_2} \cdot y_j^{-\sigma_1 \times s_j.v - \sigma_2}, x_i.y_j\right) \cdot e\left(g_2^{\lambda_{i,1}+\lambda_{i,2}-\mu_{j,1}-\mu_{j,2}}, x_i.y_j\right)$$
$$= e\left(x_i^{\sigma_1 \times r_i.u + \sigma_2}, x_i\right) \cdot e\left(y_j^{-\sigma_1 \times s_j.v - \sigma_2}, x_i\right) \cdot e\left(x_i^{\sigma_1 \times r_i.u + \sigma_2}, y_j\right) \cdot e\left(y_j^{-\sigma_1 \times s_j.v - \sigma_2}, y_j\right) \cdot e\left(g_2^{\lambda_{i,1}+\lambda_{i,2}-\mu_{j,1}-\mu_{j,2}}, x_i.y_j\right)$$
$$= e(x_i, x_i)^{\sigma_1 \times r_i.u + \sigma_2} \cdot e(y_j, x_i)^{-\sigma_1 \times s_j.v - \sigma_2} \cdot e(x_i, y_j)^{\sigma_1 \times r_i.u + \sigma_2} \cdot e(y_j, y_j)^{-\sigma_1 \times s_j.v - \sigma_2} \cdot e\left(g_2^{\lambda_{i,1}+\lambda_{i,2}-\mu_{j,1}-\mu_{j,2}}, x_i\right) \cdot e\left(g_2^{\lambda_{i,1}+\lambda_{i,2}-\mu_{j,1}-\mu_{j,2}}, y_j\right)$$
$$= e(x_i, x_i)^{\sigma_1 \times r_i.u + \sigma_2} \cdot e(x_i, y_j)^{-\sigma_1 \times s_j.v - \sigma_2} \cdot e(x_i, y_j)^{\sigma_1 \times r_i.u + \sigma_2} \cdot e(y_j, y_j)^{-\sigma_1 \times s_j.v - \sigma_2} \cdot e(g_2, x_i)^{\lambda_{i,1}+\lambda_{i,2}-\mu_{j,1}-\mu_{j,2}} \cdot e(g_2, y_j)^{\lambda_{i,1}+\lambda_{i,2}-\mu_{j,1}-\mu_{j,2}}$$
$$= e(x_i, x_i)^{\sigma_1 \times r_i.u + \sigma_2} \cdot e(x_i, y_j)^{\sigma_1 (r_i.u - s_j.v)} \cdot e(y_j, y_j)^{-\sigma_1 \times s_j.v - \sigma_2} \cdot e(g_2, x_i)^{\lambda_{i,1}+\lambda_{i,2}-\mu_{j,1}-\mu_{j,2}} \cdot e(g_2, y_j)^{\lambda_{i,1}+\lambda_{i,2}-\mu_{j,1}-\mu_{j,2}}$$

[0081] Therefore, the test condition works out to be:-

$$\frac{e(x_i,x_i)^{\sigma_1 \times r_i.u + \sigma_2} \cdot e(x_i,y_j)^{\sigma_1 (r_i.u - s_j.v)} \cdot e(y_j,y_j)^{-\sigma_1 \times s_j.v - \sigma_2} \cdot e(g_2,x_i)^{\lambda_{i,1}+\lambda_{i,2}-\mu_{j,1}-\mu_{j,2}} \cdot e(g_2,y_j)^{\lambda_{i,1}+\lambda_{i,2}-\mu_{j,1}-\mu_{j,2}}}{e(x_i,x_i)^{\sigma_1 \times r_i.u + \sigma_2} \cdot e(y_j,y_j)^{-\sigma_1 \times s_j.v - \sigma_2} \cdot e(g_2,x_i)^{\lambda_{i,1}+\lambda_{i,2}-\mu_{j,1}-\mu_{j,2}} \cdot e(g_2,y_j)^{\lambda_{i,1}+\lambda_{i,2}-\mu_{j,1}-\mu_{j,2}}} = e(x_i, y_j)^{\sigma_1 (r_i.u - s_j.v)}.$$

[0082] When $r_i.u$ is equal to $s_j.v$, $e(x_i, y_j)^{\sigma_1 (r_i.u - s_j.v)} = e(x_i, y_j)^0 = 1$ where 1 is the identity element of G_T. Therefore, if test condition $\frac{e(A_{i,3} \cdot B_{j,3}^{-1}, A_{i,4} \cdot B_{j,4})}{A_{i,8} \cdot B_{j,8}^{-1} \cdot e\left(A_{i,1} \cdot B_{j,1}^{-1}, g_2\right) \cdot e(A_{i,7}, B_{j,6}^{\kappa_B / \tau_A}) \cdot e(B_{j,7}^{-\kappa_A / \tau_B}, A_{i,6}) \cdot e(A_{i,2} \cdot B_{j,2}^{-1}, A_{i,5} \cdot B_{j,5})} = 1$ holds, un-trusted server will determine that $r_i.u$ matches $s_j.v$.

[0083] This encryption method is advantageous as it is probabilistic. This is because integers $\lambda_{i,1}$ and $\lambda_{i,2}$ as well as element x_i are randomly selected each time $r_i.u$ is encrypted, and integers $\mu_{j,1}$ and $\mu_{j,2}$ as well as element y_j are randomly selected each time $s_j.v$ is encrypted. Therefore, for $r_i.u$ or $s_j.v$ encrypted at different times, the encrypted data A_i and B_j will be different. Despite the fact that every encryption of $r_i.u$ and $s_j.v$ will result in different

encrypted data A_i and B_j, the test condition $\frac{e\left(A_{i,3}\cdot B_{j,3}^{-1}, A_{i,4}\cdot B_{j,4}\right)}{A_{i,8}\cdot B_{j,8}^{-1}\cdot e\left(A_{i,1}\cdot B_{j,1}^{-1}, g_2\right)\cdot e\left(A_{i,7}, B_{j,6}^{\kappa_B/\tau_A}\right)\cdot e\left(B_{j,7}^{-\kappa_A/\tau_B}, A_{i,6}\right)\cdot e\left(A_{i,2}\cdot B_{j,2}^{-1}, A_{i,5}\cdot B_{j,5}\right)} = 1$ will always determine correctly whether $r_i.u$ is equal to $s_j.v$. This is due to the specific structure of the test condition, encrypted data A_i and B_j and token τ_{AB}, which results in the condition $e(x_i, y_j)^{\sigma_1(r_i.u - s_j.v)} = 1$, so that integer σ_1, element x_i and element y_j will be neutralized when $r_i.u$ is equal to $s_j.v$.

[0084] Another advantage of the encryption method is that the un-trusted server can perform the equijoin only after receiving token τ_{AB} from the client machine. Further still, token τ_{AB} can only be used for an equijoin of table R and table S on attribute u in R and attribute v in S. The token cannot be used to join R and S on any other attributes, nor for joining other tables.

[0085] Figure 4 shows a system for implementing the method in accordance with the preferred embodiment and shows the data exchange between client machine 401 and un-trusted server 402. As shown, client machine 401 sends p, g_1 and g_2 as a public key to un-trusted sever 402. This public key is stored in un-trusted server 402. Un-trusted server 402 requires this public key to perform the test condition. Client machine 401 does the encryption of data and stores encrypted data A and B in un-trusted server 402. To test whether two data values are equal, the token $\tau_{AB} = (\kappa_B/\tau_A, -\kappa_A/\tau_B)$ is sent to un-trusted server 402 so that un-trusted server 402 can perform the test condition. If the test condition holds, un-trusted server 402 would have determined that the two data values are equal, all the while not being privy to the actual values of the two data values.

[0086] It will be apparent that various other modifications and adaptations of the application will be apparent to the person skilled in the art after reading the foregoing disclosure without departing from the spirit and scope of the application and it is intended that all such modifications and adaptations come within the scope of the appended claims.

[0087] In the application, unless specified otherwise, the terms "comprising", "comprise", and grammatical variants thereof, are intended to represent "open" or "inclusive" language such that they include recited elements but also permit inclusion of additional, non-explicitly recited elements.

CLAIMS

1. A method for determining whether a first encrypted data of a first data value is equal to a second encrypted data of a second data value, the method comprising the steps of :-

composing a first cyclic group, the first cyclic group comprising a plurality of elements;

composing a second cyclic group, the second cyclic group comprising a plurality of elements including a first element;

applying a mathematical operation to the first cyclic group to map elements of the first cyclic group to one of the elements in the second cyclic group;

randomly selecting a second element from the first cyclic group;

producing the first encrypted data by mapping the second element and the first data value into one or more elements of the first cyclic group;

randomly selecting a third element from the first cyclic group;

producing the second encrypted data by mapping the third element and the second data value into one or more elements of the first cyclic group; and

performing a test condition by applying the mathematical operation to the first encrypted data and the second encrypted data to obtain a fourth element in the second cyclic group, wherein the fourth element is equal to the first element when the first data value is equal to the second data value.

2. The method of claim 1 further comprising the steps of randomly selecting integers to form a secret key; and generating a token, the token being a function of the secret key; and

wherein the step of producing the first encrypted data comprises the step of mapping the second element, the first data value and the secret key into one or more elements of the first cyclic group; and

wherein the step of producing the second encrypted data comprises the step of mapping the third element, the second data value and the secret key into one or more elements of the first cyclic group; and

wherein the step of performing a test condition comprises the step of applying the mathematical operation to the first encrypted data, the second encrypted data and the token.

3. The method of claim 1 or claim 2 wherein the mathematical operation is a bilinear mapping operation.

4. The method of any one of the preceding claims wherein the first element is an identity element of the second cyclic group.

5. A method for determining which probabilistically encrypted values in a first set is equal to the probabilistically encrypted values in a second set, the method comprising the steps of :-

extracting a first data value from the first set;

extracting a second data value from the second set; and

determining whether a first encrypted data of the first data value is equal to a second encrypted data of the second data value by using the method as claimed in any one of claims 1 to 4.

6. A method for determining which probabilistically encrypted values in a first table is equal to the probabilistically encrypted values in a second table, the method comprising the steps of :-

extracting a first record from the first table, the first record having a first attribute with a first data value;

extracting a second record from the second table, the second record having a second attribute with a second data value; and

determining whether a first encrypted data of the first data value is equal to a second encrypted data of the second data value by using the method as claimed in any one of claims 1 to 4.

7. A system for determining whether a first encrypted data of a first data value is equal to a second encrypted data of a second data value, the system comprising :-

a client machine, the client machine configured to :-

compose a first cyclic group, the first cyclic group comprising a plurality of elements;

compose a second cyclic group, the second cyclic group comprising a plurality of elements including a first element;

apply a mathematical operation to the first cyclic group to map elements of the first cyclic group to one of the elements in the second cyclic group;

randomly select a second element from the first cyclic group;

produce the first encrypted data by mapping the second element and the first data value into one or more elements of the first cyclic group;

randomly select a third element from the first cyclic group; and

produce the second encrypted data by mapping the third element and the second data value into one or more elements of the first cyclic group;

and a server, the server configured to :-

receive the first encrypted data and the second encrypted data from the client machine; and

perform a test condition by applying the mathematical operation to the first encrypted data and the second encrypted data to obtain a fourth element in the second cyclic group, wherein the fourth element is equal to the first element when the first data value is equal to the second data value.

8. The system of claim 7 wherein the client machine is further configured to :-

randomly select integers to form a secret key;

generate a token, the token being a function of the secret key;

produce the first encrypted data by mapping the second element, the first data value and the secret key into one or more elements of the first cyclic group; and

produce the second encrypted data by mapping the third element, the second data value and the secret key into one or more elements of the first cyclic group.

9. The system of claim 8 wherein the server is further configured to :-

receive the token from the client machine;

apply the mathematical operation to the first encrypted data, the second encrypted data and the token to obtain a fourth element in the second cyclic group, wherein the fourth element is equal to the first element when the first data value is equal to the second data value.

10. The system of any one of claims 7 to 9 wherein the mathematical operation is a bilinear mapping operation.

11. The system of any one of claims 7 to 10 wherein the first element is an identity element of the second cyclic group.

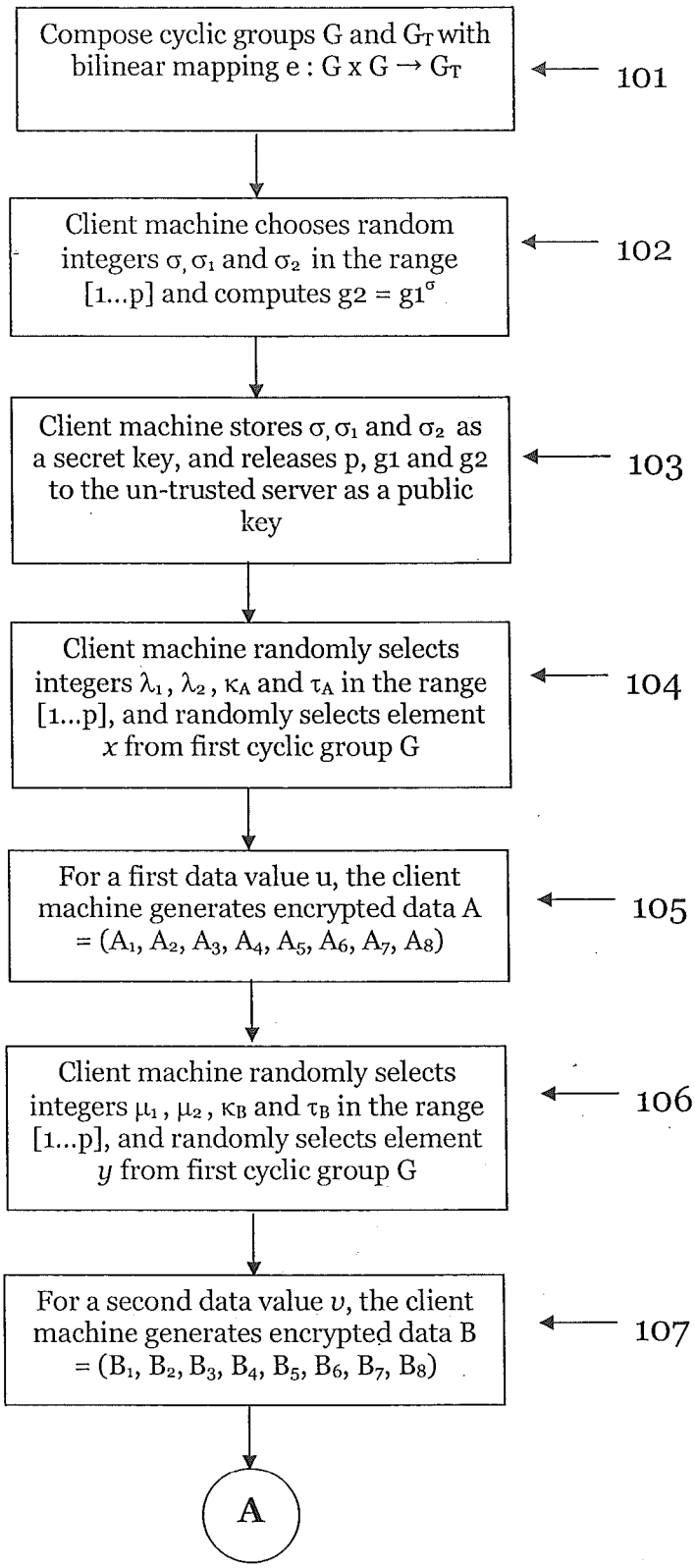
Compose cyclic groups G and G_T with bilinear mapping e : G x G → G_T
101
Client machine chooses random integers σ, σ_1 and σ_2 in the range [1...p] and computes g2 = $g1^{\sigma}$
102
Client machine stores σ, σ_1 and σ_2 as a secret key, and releases p, g1 and g2 to the un-trusted server as a public key
103
Client machine randomly selects integers λ_1, λ_2, κ_A and τ_A in the range [1...p], and randomly selects element *x* from first cyclic group G
104
For a first data value u, the client machine generates encrypted data A = (A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8)
105
Client machine randomly selects integers μ_1, μ_2, κ_B and τ_B in the range [1...p], and randomly selects element *y* from first cyclic group G
106
For a second data value υ, the client machine generates encrypted data B = (B_1, B_2, B_3, B_4, B_5, B_6, B_7, B_8)
107
A

Figure 1

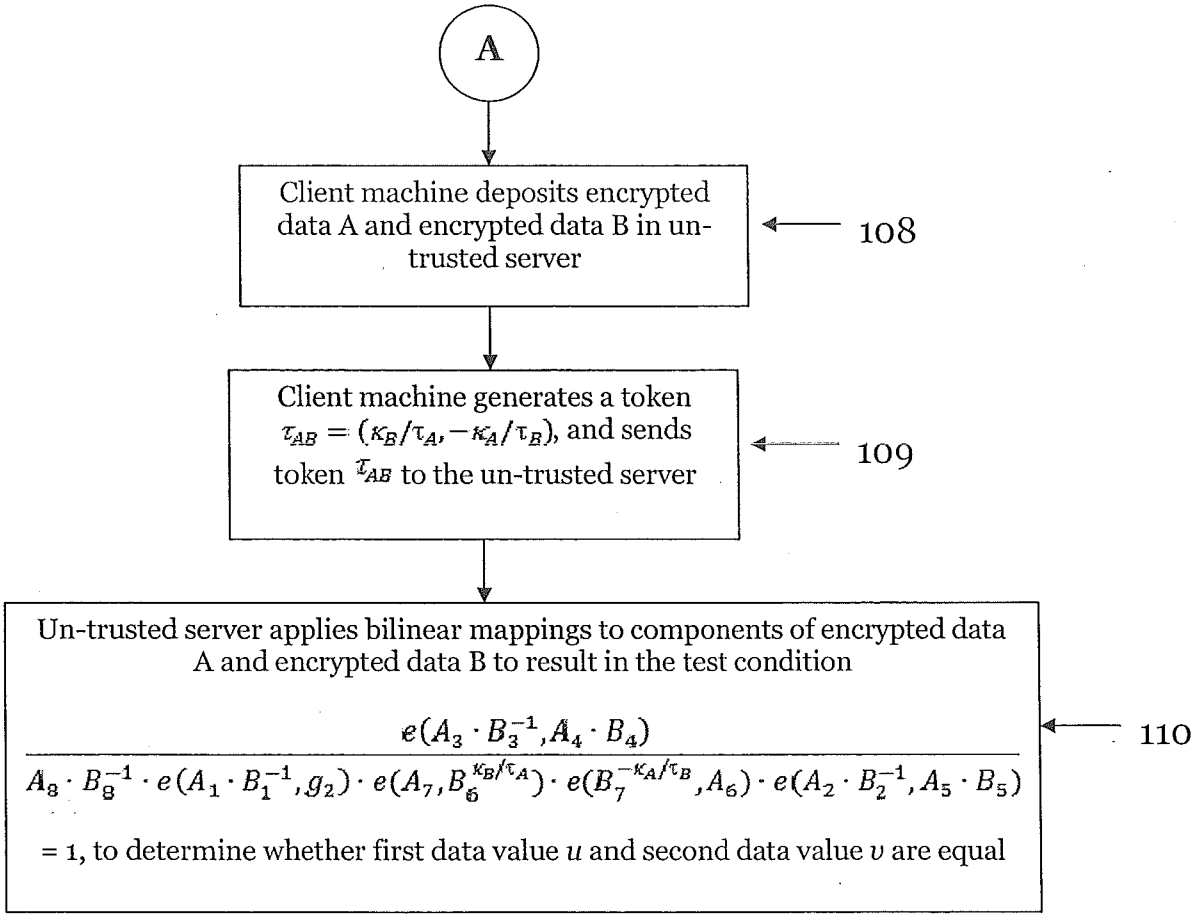

Figure 1
(continue)

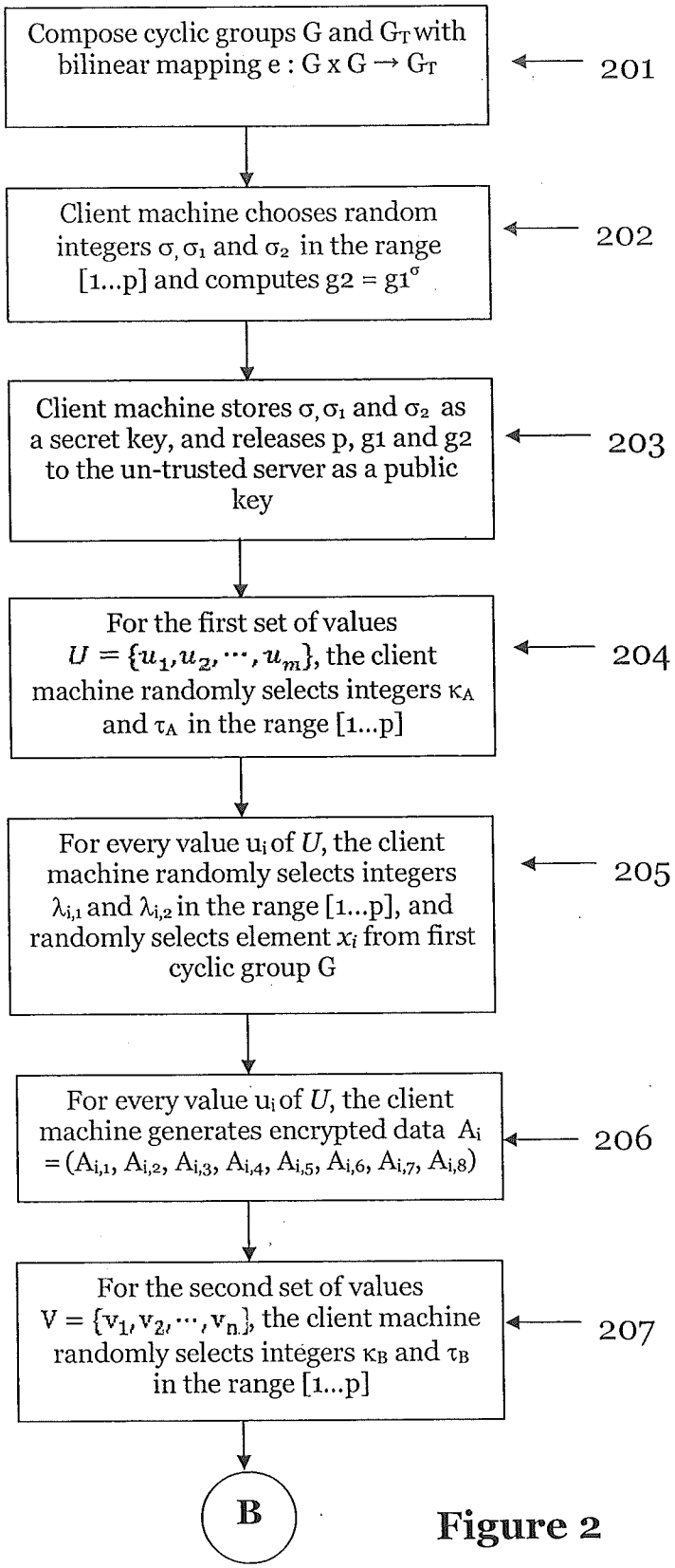
Compose cyclic groups G and G_T with bilinear mapping e : G x G → G_T
201
Client machine chooses random integers σ, σ_1 and σ_2 in the range [1...p] and computes g2 = $g1^{\sigma}$
202
Client machine stores σ, σ_1 and σ_2 as a secret key, and releases p, g1 and g2 to the un-trusted server as a public key
203
For the first set of values $U = \{u_1, u_2, \cdots, u_m\}$, the client machine randomly selects integers κ_A and τ_A in the range [1...p]
204
For every value u_i of U, the client machine randomly selects integers $\lambda_{i,1}$ and $\lambda_{i,2}$ in the range [1...p], and randomly selects element x_i from first cyclic group G
205
For every value u_i of U, the client machine generates encrypted data $A_i = (A_{i,1}, A_{i,2}, A_{i,3}, A_{i,4}, A_{i,5}, A_{i,6}, A_{i,7}, A_{i,8})$
206
For the second set of values $V = \{v_1, v_2, \cdots, v_n\}$, the client machine randomly selects integers κ_B and τ_B in the range [1...p]
207
B

Figure 2

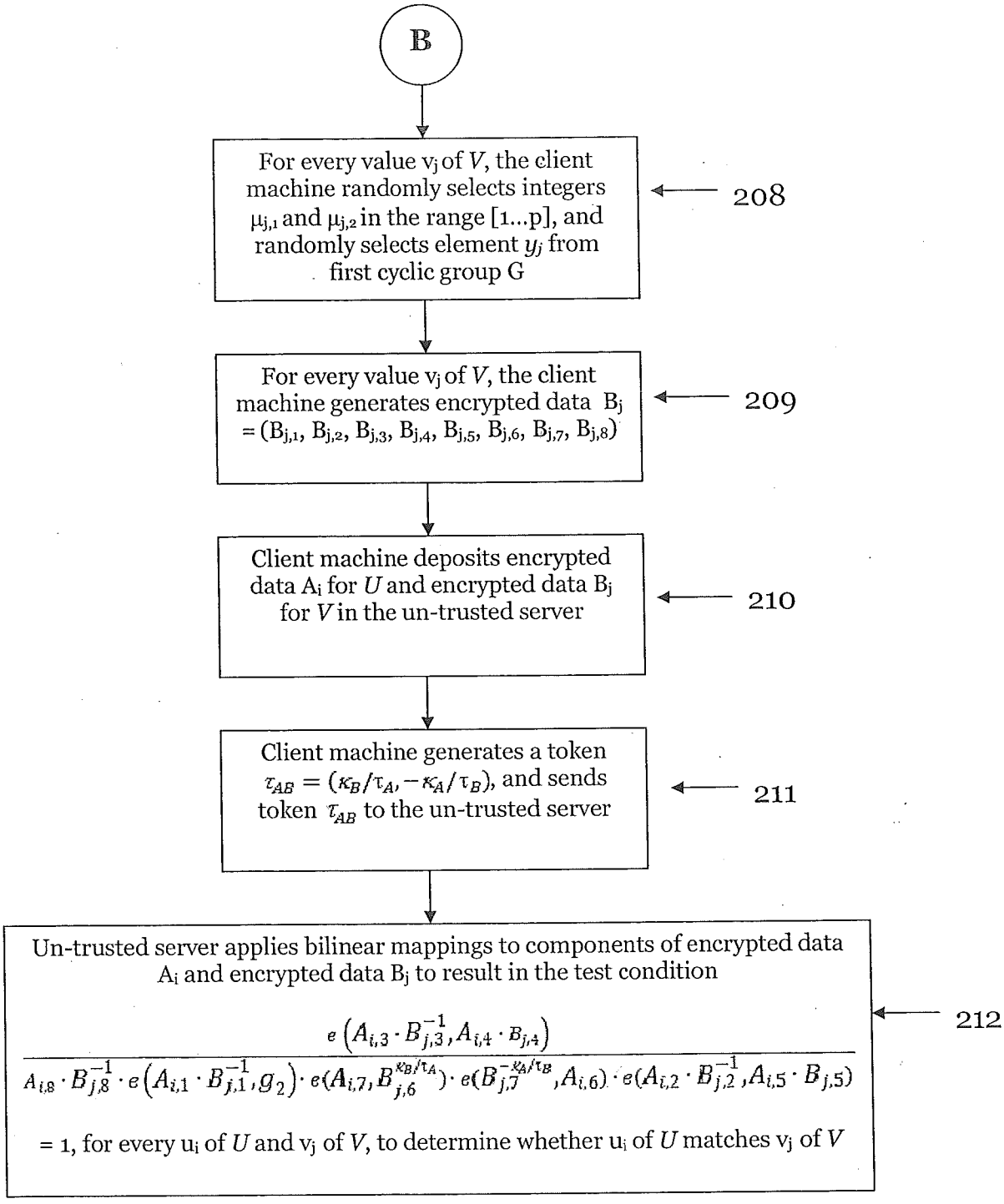

Figure 2 (continue)

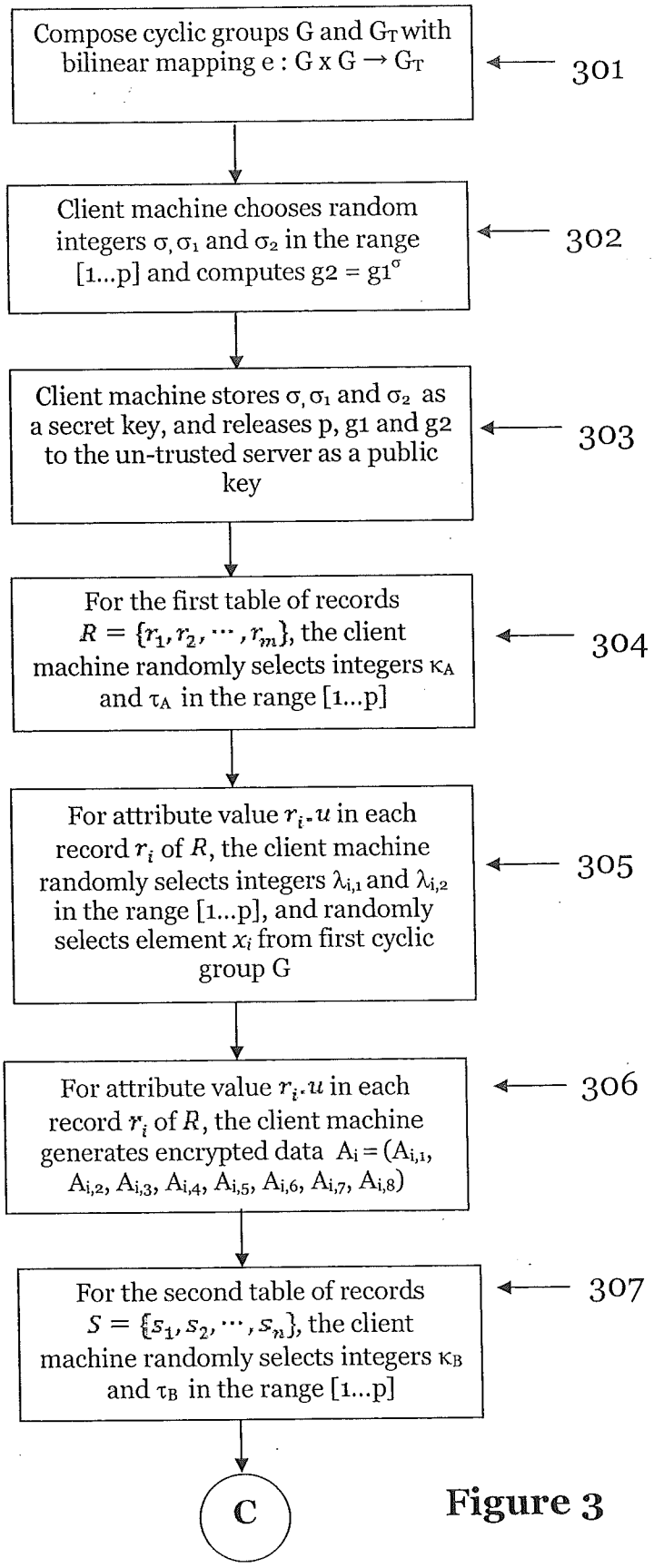
Compose cyclic groups G and G_T with bilinear mapping e : G x G → G_T
301
Client machine chooses random integers σ, σ_1 and σ_2 in the range [1...p] and computes g2 = $g1^{\sigma}$
302
Client machine stores σ, σ_1 and σ_2 as a secret key, and releases p, g1 and g2 to the un-trusted server as a public key
303
For the first table of records $R = \{r_1, r_2, \cdots, r_m\}$, the client machine randomly selects integers κ_A and τ_A in the range [1...p]
304
For attribute value $r_i.u$ in each record r_i of R, the client machine randomly selects integers $\lambda_{i,1}$ and $\lambda_{i,2}$ in the range [1...p], and randomly selects element x_i from first cyclic group G
305
For attribute value $r_i.u$ in each record r_i of R, the client machine generates encrypted data $A_i = (A_{i,1}, A_{i,2}, A_{i,3}, A_{i,4}, A_{i,5}, A_{i,6}, A_{i,7}, A_{i,8})$
306
For the second table of records $S = \{s_1, s_2, \cdots, s_n\}$, the client machine randomly selects integers κ_B and τ_B in the range [1...p]
307
C

Figure 3

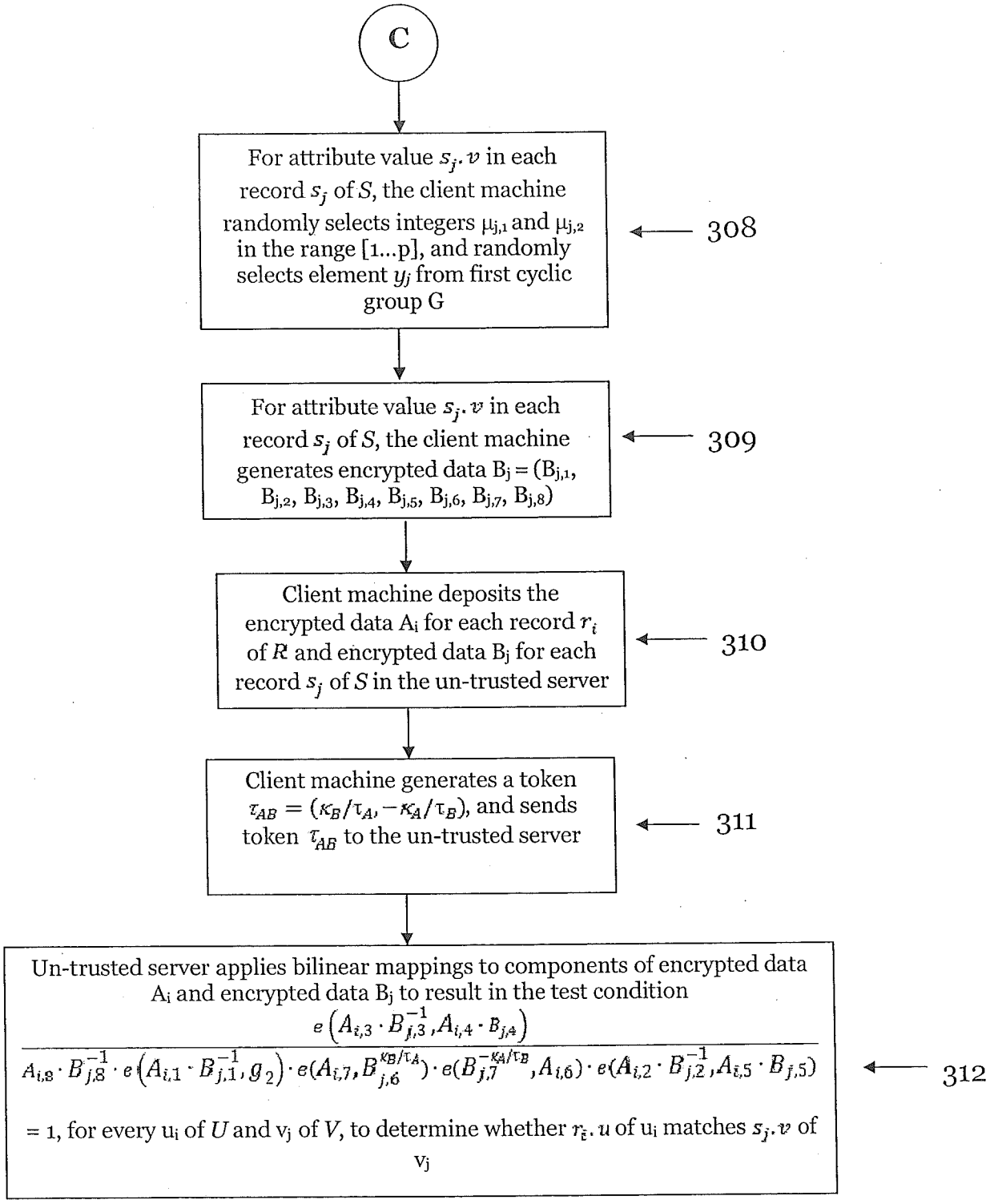

Figure 3 (continue)

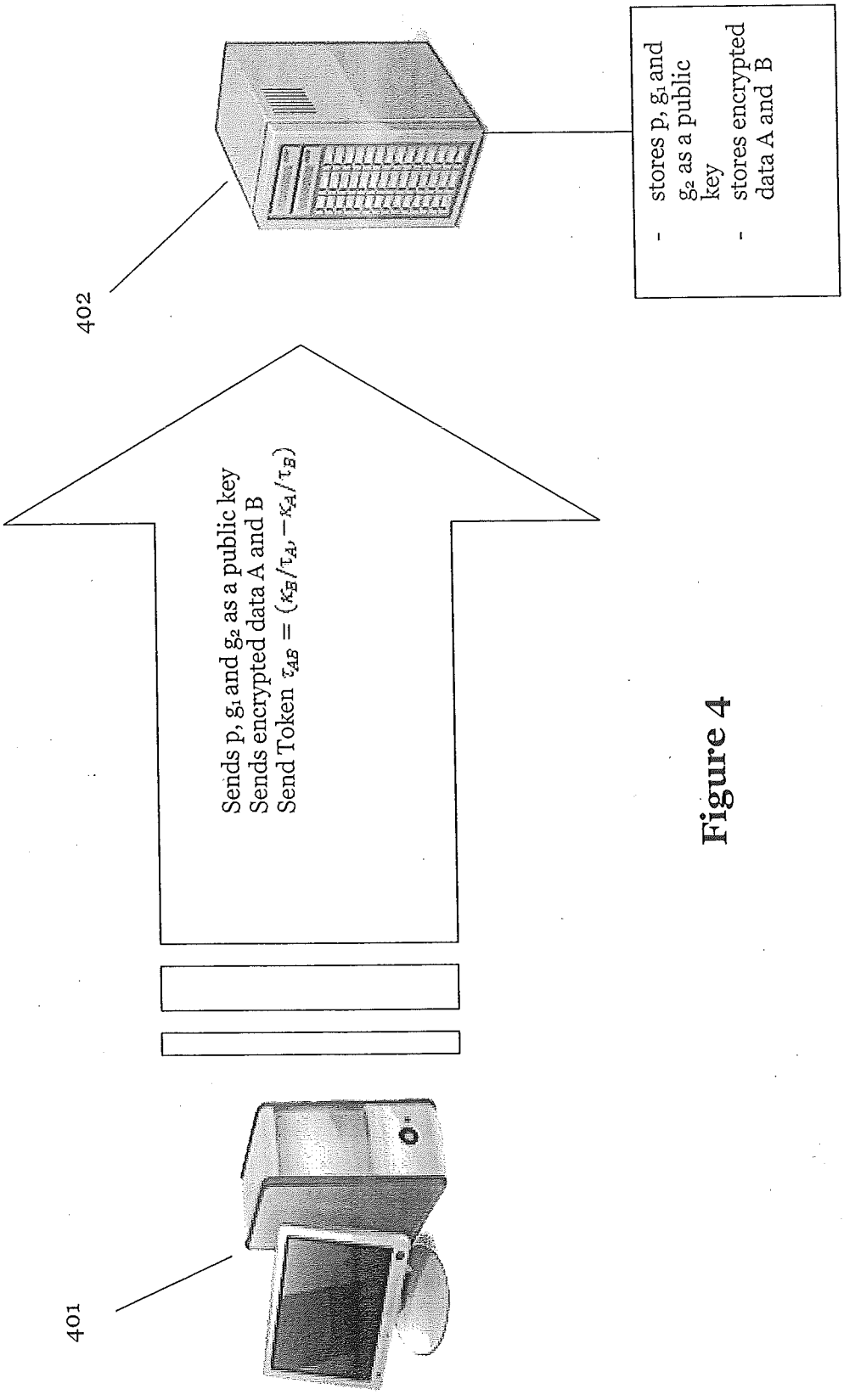
401
Sends p, g1 and g2 as a public key
Sends encrypted data A and B
Send Token $\tau_{AB} = (\kappa_B/\tau_A, -\kappa_A/\tau_B)$
402
- stores p, g1 and g2 as a public key
- stores encrypted data A and B

Figure 4

INTERNATIONAL SEARCH REPORT

International application No.
PCT/SG2014/000590

A. CLASSIFICATION OF SUBJECT MATTER

H04L 9/28 (2006.01) G06F 21/10 (2013.01)

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

ESPACE (Applicant/Inventor search); GOOGLE (Singapore Management University, probabilistic encryption, cyclic group); GOOGLE SCHOLAR (Inventor search; probabilistic, encryption, cyclic group, match, and similar terms); GOOGLE PATENTS, THE LENS, WPI, EPODOC, NPL (probabilistic, encryption, cyclic group, match, and similar terms);

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
	Documents are listed in the continuation of Box C	

[X] Further documents are listed in the continuation of Box C

[X] See patent family annex

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
9 February 2015	09 February 2015
Name and mailing address of the ISA/AU	**Authorised officer**
AUSTRALIAN PATENT OFFICE PO BOX 200, WODEN ACT 2606, AUSTRALIA Email address: pct@ipaustralia.gov.au	Andrew Ellett AUSTRALIAN PATENT OFFICE (ISO 9001 Quality Certified Service) Telephone No. 0262256120

INTERNATIONAL SEARCH REPORT C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		International application No. **PCT/SG2014/000590**
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	G. FUCHSBAUER, et al.; "Proofs on Encrypted Values in Bilinear Groups and an Application to Anonymity of Signatures"; Pairing-Based Cryptography – Pairing 2009; Lecture Notes in Computer Science Volume 5671, 2009, pp 132-149;	1-11
A	J.L. MASSEY; "Logarithms in Finite Cyclic Groups – Cryptographic Issues"; Proc. 4th Benelux Symposium on Information Theory; 1983;	1-11
A	US 2013/0083921 A1 (FUJISAKI) 04 April 2013	1-11

INTERNATIONAL SEARCH REPORT Information on patent family members	International application No. **PCT/SG2014/000590**

This Annex lists known patent family members relating to the patent documents cited in the above-mentioned international search report. The Australian Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

Patent Document/s Cited in Search Report		Patent Family Member/s	
Publication Number	**Publication Date**	**Publication Number**	**Publication Date**
US 2013/0083921 A1	04 April 2013	US 8897442 B2	25 Nov 2014
		CN 103004129 A	27 Mar 2013
		EP 2597812 A1	29 May 2013
		JP 5466763 B2	09 Apr 2014
		KR 20130024931 A	08 Mar 2013
		WO 2012011564 A1	26 Jan 2012

End of Annex

Due to data integration issues this family listing may not include 10 digit Australian applications filed since May 2001.